Foundation Model for Phase Field Dynamics

Pascal Bärtschi
D-BSSE, ETH Zurich
pbaertschi@ethz.ch

***Abstract*—This work investigates the development of a foundation model for phase field dynamics using a Fourier Neural Operator (FNO) to approximate solutions of the Allen-Cahn equation. The Allen-Cahn equation describes phase separation in binary systems and is parameterized by ϵ, which controls interface width. A dataset covering multiple ϵ values and diverse initial conditions was generated to train the model. The architecture integrates time and parameter embeddings through conditional batch normalization and is trained with an All2All strategy to improve generalization. Results demonstrate that while the model generalizes well in-distribution, challenges arise in extrapolating to unseen ϵ values, particularly for sharper transitions.**

***Index Terms*—Fourier Neural Operator, Allen-Cahn Equation, Phase Field Dynamics, Deep Learning, Neural PDE Solvers, Conditional Batch Normalization**

I. INTRODUCTION

In this project, we aim to develop a neural foundation model to approximate solutions of the Allen-Cahn equation, a fundamental partial differential equation in materials science. The equation describes phase separation dynamics in binary systems and is given by:

$$\frac{\partial u}{\partial t} = \Delta_x u - \frac{1}{\epsilon^2}(u^3 - u),$$

where $u = u(x,t)$ represents the phase variable, ϵ is a parameter controlling interface width, and Δ_x is the Laplacian operator. The equation is solved over $t \in [0,1]$ and $x \in [-1,1]$, with periodic boundary conditions.

Our goal is to construct a neural solver capable of learning the full dynamics of the Allen-Cahn equation across varying parameter regimes (ϵ) and initial conditions. By embedding both temporal and parameter dependencies into the model architecture, we investigate its ability to generalize to unseen initial conditions and challenging scenarios, such as small ϵ values leading to sharp interface dynamics.

II. METHODS

1) Data Generation: To investigate the ability of the neural model to generalize across various regimes, datasets were generated under the following conditions:

- **Different ϵ values:** Training datasets were created for multiple values of ϵ (e.g., 0.1, 0.07, 0.05, 0.02) to capture the dynamic behavior of the Allen-Cahn equation across a spectrum of phase separation regimes. Additional out-of-distribution (ooD) datasets were generated with ϵ values outside the training range (e.g., 0.15, 0.03, 0.01) to evaluate extrapolation capabilities.
- **Various initial condition types:** A diverse set of initial conditions was used to assess generalization, including:
  - *Fourier series* with random coefficients to create smooth initial states.
  - *Gaussian Mixture Models* with random means, variances, and weights were sampled to generate complex but smooth profiles.
  - *Piecewise linear functions* with random breakpoints, occasionally incorporating discontinuities, were used to create sharp transitions.

Each dataset consisted of 5 temporal snapshots (e.g. $\{0, 0.0025, 0.005, 0.0075, 0.01\}$) of the solution generated by solving the Allen-Cahn equation using a finite-difference-based numerical solver (`scipy.integrate.solve_ivp`). The generated datasets were normalized to $[-1, 1]$ to align with the neural network's input range and enforced to adhere to the periodic boundary conditions. For every combination of epsilon and initial condition type, 100 samples were generated.

2) Model and Training: The skeleton model architecture is a Fourier Neural Operator (FNO) with three convolution layers as proposed by Li et al. (2020) [1]. As the simulated dynamics of the Allen Cahn equation are both time-dependent and influenced by epsilon in terms of trajectory smoothness, this architecture was extended. Both parameters were added to the input as an additional channel to grid and input trajectory and both individually embedded via Conditional-Batch Normalization:

$$\text{output} = \gamma(t)\frac{x-\mu}{\sqrt{\sigma^2+\epsilon}} + \beta(t), \quad \gamma(t), \beta(t) = \text{MLP}(t)$$

. In every layer these embeddings were added to the embeddings of the convolution layers (i.e., spectral plus spatial convolution). To train these embedding optimally and leverage the full power of the time evolution information, All2All training as described by Raonić et al. (2023) [2] is used. Note that all time combinations, were combined with all possible epsilon values. This lead to training with 12'000 samples. The model was trained on a Colab GPU with curriculum training (i.e., introducing new epsilon values after 20 epochs) for 120 epochs with a batch size of 32. The loss is optimized with Adam with 10^{-3} learning rate, no weight decay and a schedular that decays the learning rate by 0.5 if the validation loss has not improved since 10 epochs. The loss function used is:

$$\begin{aligned}\mathcal{L} = &\|\mathbf{u}_{\text{pred}} - \mathbf{u}_{\text{true}}\|_2^2 \\ &+ \lambda_1 \|u(x_1, t) - u(x_M, t)\|_2^2 \\ &+ \lambda_2 \|\Delta_x \mathbf{u}_{\text{pred}}\|_2^2,\end{aligned}$$

where the first term is the MSE-loss with two regularization terms: the first one to penalize deviations from periodic boundary conditions and the second to penalize unsmooth predictions (i.e. deviations from physical reality) with $\lambda_{1,2}$ set to 0.05.

III. Results

1) Generated data: Figure 1 shows an example of how the evolution of a 1D domain governed by the Allen-Cahn equation looks like. With decreasing epsilon one observes overall sharper transitions and faster decrease of smoothness with time.

Furthermore, figure 2 indicates that the decrease of smoothness is exponential. In this work the empirical variance of the trajectories is taken as a proxy for smoothness. It is assumed that sharper, less smooth transitions display higher trajectory variance. Figure 2 shows that higher epsilons lead to faster convergence to the steady states of the domain, indicated by stagnating variance. Furthermore, smaller epsilons show stronger exponential growth of the variance, also showing that the sharper transitions arise faster. Note that this trend can be observed across all initial condition type on different scales.

2) Model performance: The model converges after 120 epochs, showing interesting behavior of the training loss curve (Figure 3: Due to the curriculum learning the training loss spikes up as the loss landscape changes (i.e., new epsilon introduced) to then find a new local minimum. This way the model can learn the complexity of the true dynamics, rather than overfitting to irrelevant patterns early in the optimization process. Note that $\lambda_{1,2}$ could not be set far above 0.05, as this led to an underfit model with all combinations of hyperparameters tried.

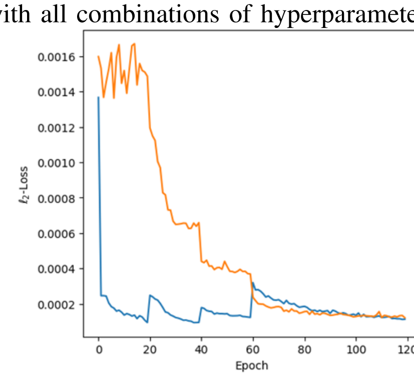

Fig. 3: Training and validation loss in model training

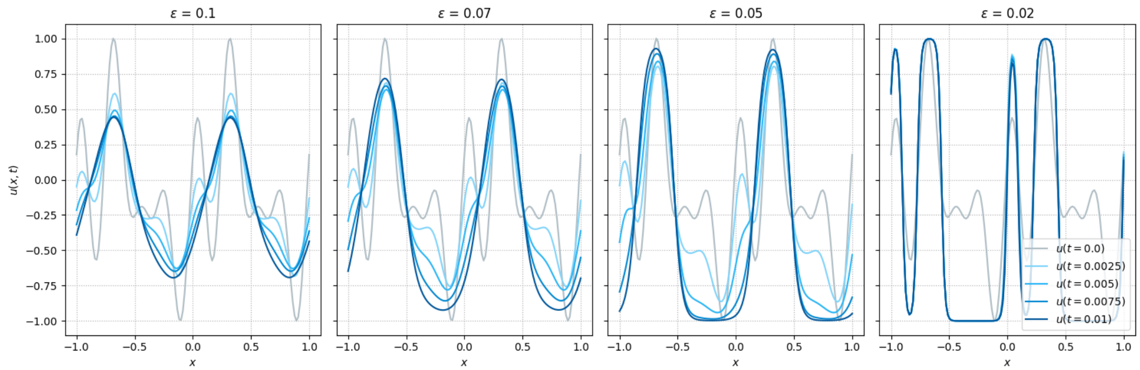

Fig. 1: Evolving Allen Cahn dynamics with different epsilons, initialized as a fourier series.

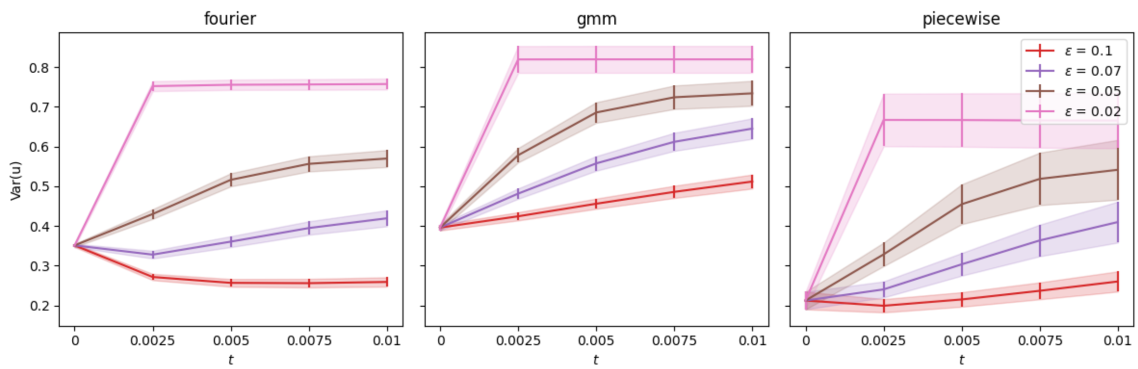

Fig. 2: Evolution of empirical variance of trajectories governed by the Allen-Cahn equation under different epsilons over time.

Tables I, II and III show the average relative L2 error obtained across different epsilons and time steps for the in- and out-of-distribution test sets. We can observe that the errors on the ooD test set are around 10% higher for the majority of time steps and epsilons. For the inD test set errors are similar (for the error range of this work) across epsilons and time steps. The ooD data shows a similar pattern for time, however, the epsilon value that needed to be interpolated (i.e., 0.03) by the model is remarkably lower than for extrapolated values. Specifically, it is only one third of $\epsilon = 0.01$ (i.e., extrapolation to sharper transitions) and half to $\epsilon = 0.15$ i.e., extrapolation to smoother dynamics), respectively.

$\boldsymbol{\ell_2(t)}$	0.0025	0.0050	0.0075	0.0100
ℓ_2 inD	5.14%	4.14%	3.89%	4.47%
ℓ_2 ooD	15.81%	14.81%	14.28%	14.22%

TABLE I: Error in- and out-of-distribution over time

$\boldsymbol{\ell_2(\epsilon)}$	$\epsilon = 0.1$	$\epsilon = 0.07$	$\epsilon = 0.05$	$\epsilon = 0.02$
ℓ_2 inD	4.40%	3.82%	4.67%	4.75%

TABLE II: Error in-distribution at different ϵ

$\boldsymbol{\ell_2(\epsilon)}$	$\epsilon = 0.01$	$\epsilon = 0.03$	$\epsilon = 0.15$
ℓ_2 ooD	21.10%	8.91%	14.34%

TABLE III: Error out-of-distribution at different ϵ

3) Generalization properties: Looking at individual predictions of dynamics to Fourier initial conditions in figure 4, one can see that the major source of error in-distribution that predictions do not align with the target in regions of smooth transitions or interfaces. Moreover, out-of-distribution we notice that the model does not generalize enough to learn sharper transitions in unseen low epsilons (i.e. 0.01, 0.03). In addition, the extrapolation to even smoother transitions lead to rather noisy, stochastic-like trajectories.

Furthermore, in figure 5 we see that error behaves similarly across initial conditions with one exception. In-distribution the spread between initial conditions is lower than out-of-distribution, where Gaussian Mixture Model IC seem to be the best learnt. For the out-of-distribution set dynamics to GMM IC are predicted remarkably better. Infact, their error range is more similar to the in-distribution set. In addition, the trend of extrapolation being harder to predict that interpolation is seen for types initial conditions tested. (Figure 5).

IV. Discussion

The generated data used for the training of the Foundation Model displays reasonable dynamics. From inspecting the structure of the Allen Cahn Equation it is straightforward that decreasing epsilon will lead to stronger nonlinear effects compared to smooth diffusion dynamics. This decrease in smoothness is visible in figure 1, where increasing epsilon leads to overall sharper transitions over time. Furthermore, we see that stronger nonlinear effects (leading to sharper transitions) also lead to faster exponential decrease of the smoothness and faster convergence to steady-states of the trajectories (figure 2. From this visual analysis, we can conclude that smaller epsilon values increase nonlinear effects on the dynamics exponentially, countering the smooth nature of diffusion encoded in the Allen-Cahn equation. Overall, the model seems to generalize reasonably well across time and different epsilons in-distribution, as the errors are more or less constant. However, in out-of-distribution testing the challenges of learning the Allen-Cahn dynamics are shown and can be summarized in three key points (Figures 4 and 5):

1) **Extrapolation is harder than interpolation**: The error of the interpolated ϵ is almost on-par with the in-distribution errors, which is not the case for extrapolated epsilons. Here extrapolating to sharper transititions (i.e., $\epsilon = 0.01$) is harder than to smoother dynamics (i.e., $\epsilon = 0.15$)
2) **High-frequency artifacts**: At smooth transitions (e.g., $\epsilon = 0.1, 0.15$ and interfaces we can see "wiggly" fits to the target. This is most likely to high-frequency components in our FNO model (embeddings in frequency domain) not being sufficiently regularized in training.
3) **High-frequency initial conditions are harder to predict**: Dynamics to piecewise or fourier initial conditions are harder to learn than to one to gmms'. This trend is particularly visible for the out-of-distribution test set and the most probable reason is that gmm's are overall smoother, leading to less complex Allen-Cahn dynamics (fewer sharp transitions and less interfaces)

To address these limitations, my first approach would be to increase the complexity of the model. For instance, by increasing depth and adding skip connections to both conditional batch normalization and the fourier embeddings. Such model may be able to fit a loss with greater regularization (i.e., higher $\lambda_{1,2}$ and learn more subtle aspects of the underlying dynamics of the Allen-Cahn equation. This might also lower the gap between in- and out-of-distribution testing, as my model likely overfit to the dynamics seen in training.

In closing, the foundation model approach taken in this work shows a lot of promising aspects, such as being able to understand the influence of time and epsilon on the dynamics to a certain extent. However, there are still some crucial development steps ahead, such as enhancing the generalization abilities to unseen epsilon values, to enhance this approach to state-of-the art error levels.

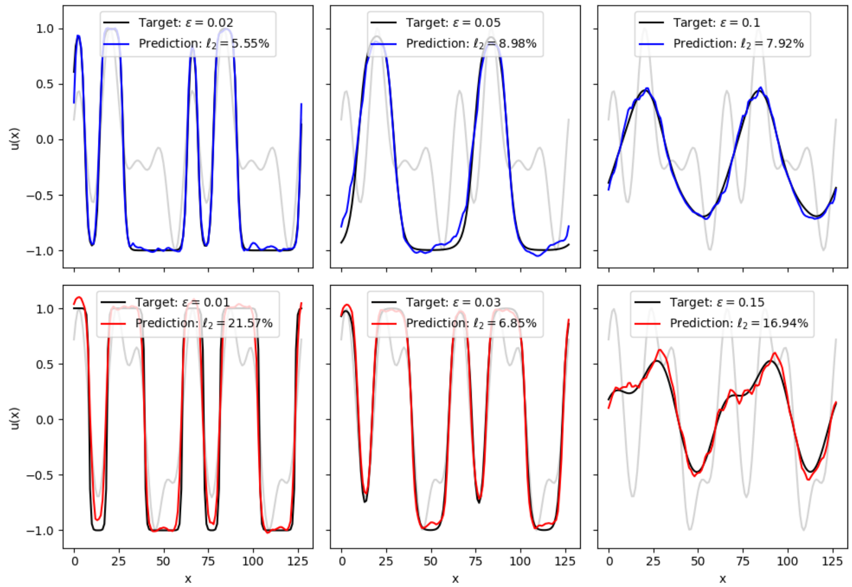

Fig. 4: Example prediction of dynamics at t=0.1 from Fourier initial conditions for inD (red) and ooD (blue) data. Initial condition trajectory in gray, targets in black and predictions in color

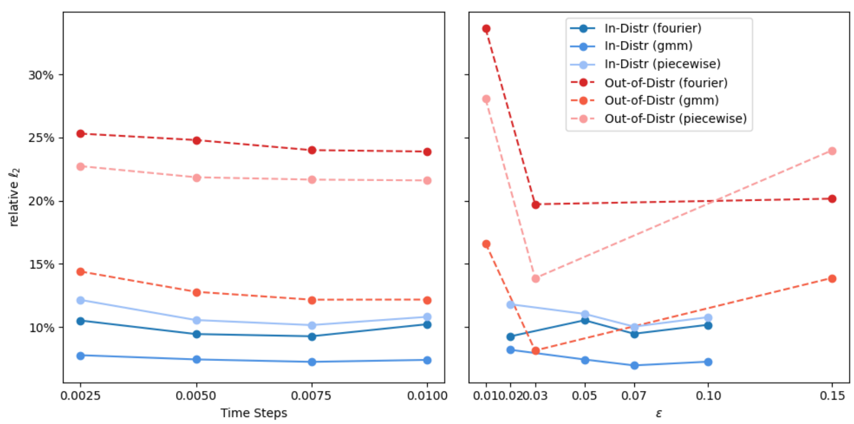

Fig. 5: Average relative ℓ_2 error across time and epsilon of inD and ooD test set, visualized by initial conditions.

References

[1] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar, "Fourier neural operator for parametric partial differential equations," *arXiv preprint arXiv:2010.08895*, 2020.

[2] B. Raonić, R. Molinaro, T. D. Ryck, T. Rohner, F. Bartolucci, R. Alaifari, S. Mishra, and E. de Bézenac, "Convolutional neural operators for robust and accurate learning of pdes," 2023. [Online]. Available: https://arxiv.org/abs/2302.01178